

**ECE 499/599 Data Compression/Information Theory
Spring 06**

Dr. Thanh Nguyen

**Homework 2
Due 04/27/06 at the beginning of the class**

Problem 4.

a)

$$H = \sum_{i=1}^5 P(a_i) \log P(a_i) = 1.817684 \text{bits}$$

b) If we sort the probabilities in descending order we can see that the two letters with the lowest probabilities are a_2 and a_4 . These will become the leaves on the lowest level of the binary tree. The parent node of these leaves will have a probability of 0.9. If we consider parent node as a letter in a reduced alphabet the it will be one of the two letters with the lowest probability: the other one being a_1 . Continuing in this manner we get the binary tree shown in Figure 1. and the code is

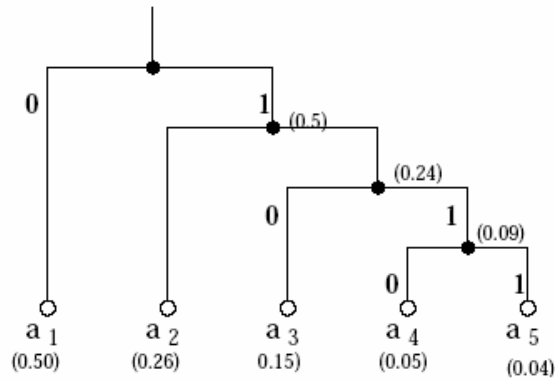


Figure 1: Huffman code for the five letter alphabet.

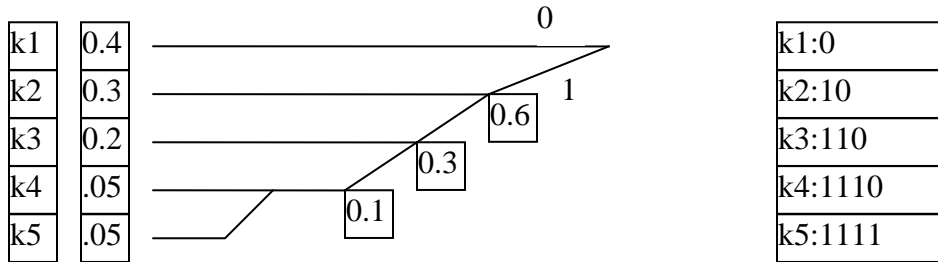
a_1	110
a_2	1111
a_3	10
a_4	1110
a_5	0

c)

$$\bar{l} = 0.15 \times 3 + 0.04 \times 4 + 0.26 \times 2 + 0.05 \times 4 + 0.5 \times 1 = 1.83 \text{bits/symbol}$$

Problem 2: Suppose you are given a task of compressing a Klingon text consisting of 10,000 letters. To your surprise, the Klingon alphabet consists of only five different letters: k1, k2, k3, k4, and k5. Furthermore, you notice that there are 4000 k1's, 3000 k2's, 2000 k3's, 500 k4's and 500 k5's. (6pts)

(a) Develop the Huffman codes for these Klingon letters.

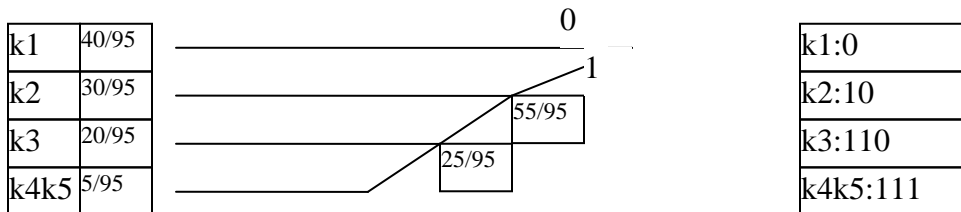


Note: There are also other valid Huffman codes!

(b) What is the average code rate?

2 bits/ symbol.

(c) Interestingly, you notice that the letters k4 and k5 always appear together as “k4k5”. Can you use this information to improve the compression ratio? If so, what are the new code and corresponding average coding rate?



Code rate = 1.8421 bits.

Problem 3: By now, you are an expert at compressing alien’s languages, the U.S. government assigns you the task of compressing Krypton language. Kryptonite are very smart (or stupid) beings, as such their entire language can be represented by sequences consisting of only three letters k1, k2, and k3. Being a compression expert that you are, you examine the

krypton text, and notice there are 7000 k1's, 2000 k2's, and 1000 k3's out of the given Krypton text containing 10,000 letters. (6pts)

(a) What is the entropy rate of this Krypton text?

1.157 bits

(b) If Huffman code is used to code the letters, what is the average code rate?

$(.7)(1) + 0.2(2) + 0.1(2) = 1.3$ bits.

(c) Due to the bandwidth limitation of inter-star communication, the US government wants you to improve your basic Huffman compression scheme. Can you modify Huffman coding to compress the Krypton text further? If so, provide an example of the new codes.

Yes, use extended Huffman code. Combine the two symbols together. An example of the new code looks like the following:

k1k1	1
k1k2	000
k2k1	001
k1k3	0100
k3k1	0101
k2k2	0111
k2k3	01101
k3k2	011000
k3k3	011001

Average code rate: = 1.165 bit per symbols.

Note that: There are also other huffman codes.

Problem 4: (bonus) Suppose you are at Las Vegas, and you notice this one peculiar game. The game is described as follows. A guy throws a peculiar dice with three faces: 1, 2, and 3. However, he would not let you know the outcome of the dice. Your job is to guess the outcome of the dice. Suppose you guess an outcome, then he would tell you *which number is not the outcome*, and gives you a second chance to guess your number again (he won't throw the dice again). If you guess wrong the second time, you have to give him \$5, on other hand if you guess right, he has to give you \$4. (2pts)

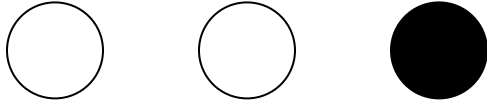
(a) What is the amount of uncertainty regarding your guesses?

Scenario A: Many of you assume that the guys always tell you the wrong number which is not the one you picked. Assume this, then the chances of you getting the right and wrong number in the first guess are $2/3$ and $1/3$, respectively. Assume you guess right the first time, then you when you switch on your second guess, you will be wrong with probability 1, or right with probability 0. Assume you guess wrong the first time, then when you switch on the second guess, you will guess right with probability 1. So if you always switch after the second guess then your probability of getting it right is:

$1/3(0) + (2/3)(1) = 2/3$. See the balls in the figure below.

If you stay with your first guess, your chance of getting it right is $1 - 2/3 = 1/3$.

Hence, the uncertainty regarding your guesses is: $H(1/3, 2/3) = 0.9$



Scenario B: Many of you also assume that the guy will tell you the wrong number regardless of whether that number is the one you picked or not. If you choose a wrong number and if with probability $1/2$ he will tell you that your number is wrong (and with probability $1/2$, he will tell you the other number is wrong) then, the chance of you getting it right when you switch is:

$$(1/3)(0) + (2/3)[(1/2)(1) + (1/2)(1/2)] = 1/2$$

Hence, $H(1/2, 1/2) = 1$ bits.

On the other hand, if he is naïve enough to think that if he always tell you that your picked number is always wrong (unless you pick the right number), then your chance of getting it right when you switch is:

$(1/3)(0) + (2/3)(1/2) = 1/3$. And hence he will gain. Well he is wrong, if you know he is using this strategy, you should not switch since your chance of getting it right is $2/3$ 😊.

(b) Would you play the game (I assume you like to win some money)?

Yes. Assume Scenario A: The average amount of money you would win is $2/3 \times 4 - 1/3 \times 5 = \1 .

No. Assume Scenario B: You lose \$1 per game on average.

